Heuristics for Joint Architecting

Maj. Todd. Wieser, USAF ▪ Maj. Gregory J. Miller, USAF ▪ Maj. Aaron Piepkorn, USAF ▪ Maj. James Kennedy, USAF ▪ Robert Mills ▪ Lt. Col. John Colombi, USAF

There is no question that Department of Defense and chairman of the Joint Chiefs of Staff directives have increased the development of operational and systems architectures. The DoD Architectural Framework (DoDAF) and its associated governing publications have provided considerable information and examples covering DoD architecture processes and products for those engaged in requirements and architecture development. But even though the guidance is good, there need to be more general heuristics to help guide those involved in this growth industry. Based on recent experiences with joint architecture development, we propose some general heuristics covering the following: the architecture team; common lexicon; process ownership; appropriate abstraction; organizational bias; level-of-war bias; and hollow-transfer activities.

The Architecture Team

The majority of architecture producers in the DoD are either government civilians or contractors. Borrowing an Army slogan, they are also often an army of one. Their levels of formal architecture education and training usually vary, and their domain knowledge of the area being modeled is usually low. Ultimately, lack of knowledge in the domain equals architecture pain. If at all possible, members of an architecture team should not only understand architecture design well, but also have real-life experience in the domain being modeled. Unfortunately, because of personnel and budget constraints, that may not be possible. Therefore, how well an architect or an architecture team develops an extended team of subject matter experts and contacts is critical to developing a useful architecture. If the architecting team makes little to no effort to seek out domain expertise when they do not have it, or if they reference only governing publications and briefs, the models produced will be poor, and the architecture will most likely not provide the benefits sought.

Common Lexicon

The lack of common terminology is quickly apparent in any joint endeavor. There are still a number of terminology differences between the Services that often confuse those outlining operational architecture inputs, controls, and outputs. An example is the terms that different Ser-

> If at all possible, members of an architecture team should not only understand architecture design well, but also have real-life experience in the domain being modeled.

vices use for a pre-execution practice. The Army and Marines often use the term "rock drill," while the Air Force primarily uses the term "rehearsal." Both mean roughly the same thing, but to Air Force personnel not familiar with Army terminology, discussion of a "rock drill" can be confusing. Therefore, use the joint dictionary in order to have a joint vocabulary. Any time definitions are needed, use joint standards and sources, such as Joint Publication 1-02, *Department of Defense Dictionary of Military and Associated Terms*. If those do not work, reference multi-Service publications. If there is no resolution, all the applicable definitions should be provided, with indication that they mean the same thing.

Process Ownership

Determining who owns the process in joint activities has long been a part of doctrinal debates and continues to influence how the Services integrate and interoperate. Even though one Service may be the lead for certain types of operations (for example, the Air Force for air superiority or the Army for the land campaign), other Services can execute the same or very similar processes within the same domain. The Navy can conduct air superiority missions and the Marines can conduct land operations. With multiple Services and commands involved, there are multiple and overlapping guidances, terminologies, and techniques. Overlapping guidance adds to the confusion when attempting to standardize common processes, especially with operational activities in joint enterprise architecture. Ideally, joint architectures need to have buy-in by all the

Wieser is assigned to Air Force Special Operations Command, Hurlburt Field, Fla. Miller, Piepkorn, and Kennedy are assigned to Headquarters, United States Air Force, Washington, D.C. Mills and Colombi are assistant professors of electrical engineering at the Air Force Institute of Technology.

	Form Approved OMB No. 0704-0188
Report Documentation Page	

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **NOV 2006**	2. REPORT TYPE	3. DATES COVERED **00-00-2006 to 00-00-2006**
4. TITLE AND SUBTITLE **Heuristics for Joint Architecting**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Air Force Special Operations Command,Hurlburt Field,FL,32544**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**		
13. SUPPLEMENTARY NOTES		
14. ABSTRACT		
15. SUBJECT TERMS		

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**5**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

major stakeholders. Unfortunately, this is not always the case. Therefore, when defining a joint process stalls, there needs to be a process owner to make firm calls so the architecture moves forward, a base standard is set, and interoperability is achieved when needed. This owner or lead agent must benevolently determine what the core activities in an operational architecture will be and how to overcome community or stakeholder differences.

Appropriate Abstraction

One of the hardest things to do when developing architecture is to define the level of abstraction. How deep in the weeds does the architect or architecture team go? DoDAF states that the "degree of granularity should be driven by the type of analysis or assessments that are of interest." But finding the right level of granularity can be very hard, and it can take multiple design iterations. If models are made at a high level only, the architect risks developing architecture that can be easily briefed to top leaders and fills program requirements, but does not answer critical questions for field operators and true stakeholders. This becomes a critical tradeoff in joint enterprise architecture that should not be quickly overlooked. The right level of abstraction highlights commonality and critical differences across Services and commands. At the same time, it also addresses operational processes in enough detail to allow informed decisions for the questions being asked. If the right level of abstraction is not chosen, the model is useless to those who need it the most. Therefore, abstract too high—the models can lie; abstract too low—one gets lost in the flow. Finding the right level of abstraction is critical in ensuring the architecture can be communicated effectively and still be useful for its intended purpose.

Organizational Bias

Within the DoD, almost everything revolves around the organization. This includes funding, identity, deployments, and other activities. Existing regulations tend to focus on job titles, roles, and responsibilities—not on key processes. When transformation is conducted, the first questions asked usually concern where people will be assigned and what organization charts will look like. It has often been said that the default method to solve a government problem is to generate a new organization. This mindset—thinking in terms of organization and jobs first—is what we call "organizational bias." DoDAF operational views are supposed to focus on activities and functions, not on organizations. As enterprise architects look across a complex environment like the DoD system of systems, they usually find it is easier to identify organizations and not underlying activities. DoD architects must realize that

FIGURE 1. Activity-Based Node Tree
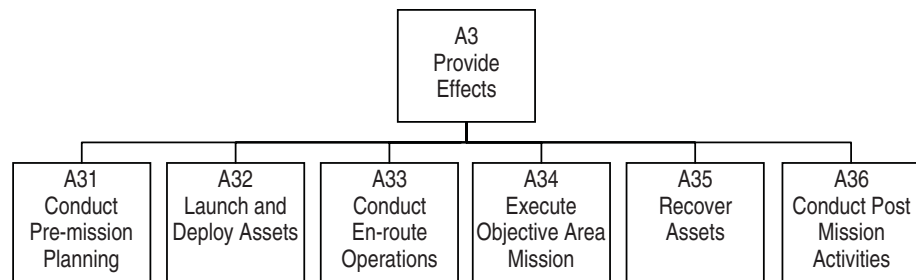
many of the publications they reference and the subject matter experts with whom they consult will tend to have this bias and will not focus on core processes. Unfiltered organizational bias can result in operational activities that are stove-piped and inefficient. This is easily seen when examining an activity node tree (OV-5) that has been heavily influenced by organizational structure. The organizational bias is often depicted as repeated boxes that identify the same or similar activity in different branches on the tree.

To illustrate, consider a notional example in which the architecture team is modeling the operational activities of an Air Force special operations organization. Assume the organization has three main functions: provide force application (direct attack on adversary forces); provide mobility (infiltration and exfiltration); and perform psychological operations (dropping leaflets and broadcasting television or radio programs). Each of these functions is performed using different assets (personnel and aircraft) but involves similar activities, such as "pre-mission planning," "launch aircraft," "conduct en-route operations," "accomplish recovery," and "conduct post-mission debriefs."

An architecture model—and more important, a mindset—that relies too much on organizational form could very easily result in an activity node tree with major branches built around each mission function and with duplicate lower-level branches. These lower-level branches may result in development of numerous tools or systems, all essentially aimed at providing the same capability. For example, three different (stove-piped) systems could very well be developed to facilitate "conduct en-route operations" for the force application, provide mobility, and perform psychological operations functions. These stove-piped systems would likely result in higher cost and reduced interoperability and flexibility.

To minimize this organizational bias, operational modeling should focus on the functions and activities to be performed, rather than on who or what unit performs them. This is illustrated in the activity node tree shown in Figure 1. Once the common processes are mapped, the truly different activities stem from the common ones and can

Druyun's home sale appears just as she advertised it: a mere coincidence that met all legal requirements. Were a violation to exist, it would be of 18 U.S.C. section 209, a law prohibiting supplementation of federal salary by a nonfederal entity. However, on the face of the information provided, some questions were bound to be raised, as evidenced by several press articles including one in the Oct. 7, 2003 *Wall Street Journal* and another in the *Washington Post* of the following day.

To serve as a comparison, take the house sale by Rep. Randy "Duke" Cunningham, R-Calif., to Mitchell Wade—a clearly illegal exchange. Cunningham sat on the House Defense Appropriations Subcommittee. Wade's company, MZM, Inc., was in defense intelligence contract work. Wade purchased Cunningham's house in November 2003 for $1,675,000. Shortly after Wade purchased the house, MZM began receiving multimillion-dollar contracts. Wade put the former Cunningham house on the market immediately after the purchase and never occupied it. It remained on the market for seven months. Wade eventually sold it for a loss of approximately $700,000.

Key differences exist between the two cases. The purchase price for Druyun's house was in line with other purchases in the same neighborhood. The fact that Druyun possessed the house for a relatively short time while making a substantial gain is irrelevant unless the August 2001 seller somehow conspired with Boeing to sell an artificially underpriced home. There is no evidence to support such a conspiracy theory. In addition, the Northern Virginia real estate market was booming at the time, and annual 10 percent increases in home values were the norm. Finally, Judy occupied the house he purchased, indicating he was buying it for himself.

The inflated purchase price of Cunningham's house, as evidenced by the subsequent sale at a significant loss, was clearly a subterfuge for bribery. Wade's failure to occupy the house and his attempt to sell it that same month further support the bribery charge.

Although the Druyun sale met all legal requirements, there is an important point to consider. Individuals occasionally find themselves in a conundrum between behavior that is legal for the individual yet may not be good for the organization. This is often referred to as an *apparent conflict of interest*. It is easy to rationalize away apparent conflicts of interest, especially if the action is not a legal violation.

Not every individual action is good for the organization. In this case, even if Druyun had not been convicted of other violations, the DoD decision to lease Boeing tankers would have been tainted by the sale of her home, even though she had cleared it with Air Force general counsel. The government does not expect individuals to take a monetary loss, but in light of the Northern Virginia housing market at the time, Druyun could have declined to sell the house to Judy based on the apparent conflict and could still have reasonably expected to make a fairly rapid sale at a comparable price to another buyer.

Individuals should guard against apparent conflicts of interest—which need not be of the magnitude of a house transaction. For example, do you meet alone with a vendor at the end of every quarter just before a big order is placed? There may be valid reasons for doing so, but the natural inclination for someone observing the behavior is to suspect that some illegal business may be going on. Instead of meeting alone, think about taking someone with you as an observer, or use it as a training opportunity for a less experienced employee. By increasing transparency in your individual activities, you may reduce the apparent conflict.

be depicted in the lower levels and branches of the tree. By keeping the focus on process, we can better ensure interoperability and minimize the amount of stove-piped system solutions. Bottom line: For the process to rule the show, organizational bias has to go.
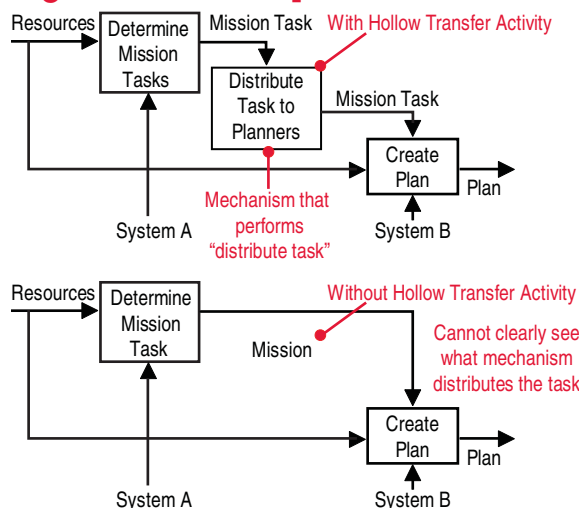
Level-of-War Bias

This bias stems from the fact that the majority of military organizations and the systems that support them are partitioned into two levels: the operational and the tactical. The operational level focuses on what, where, when, and how forces will be organized, integrated, and employed to achieve strategic goals. These are higher-order activities that primarily guide and govern the activities of the tactical level. The tactical level focuses on lower-level ac-

tivities, specifically the execution of specific missions. Organizations are formed at this level and usually report to an operational level headquarters or operating center. Although the levels are relatively easy to differentiate and understand, real processes do not restrict themselves to these human-created divisions. As network capabilities increase and organizations are pushed to transform into more streamlined and flatter entities, the lines between the tactical and operational levels blur. Viewing processes as a whole and not restricting them to operational or tactical lanes only is essential to becoming more effective and is a main aspect of many business process reengineering and Lean methodologies. Architects need to realize this and recognize when individuals speak and think with a level-of-war bias. For example, talking to someone

at a command headquarters will often center on operational-level systems and processes. Talking with those at the squadron or company level will often result in tactical-level emphasis. But unlike these conversations, operational and tactical processes do not operate in isolation, and neither should their architectures. The architect must see these biases and seek the whole picture process and then pick the right level of abstraction. Therefore, to confine the architecture to only one level of war can make the architecture poor.

Hollow Transfer Activities

Using the DoDAF, operational activities are often modeled using integrated definition (IDEF) methods, specifically the IDEF0 (pronounced IDEF-zero) function modeling method. IDEF models were originally developed for process modeling involving physical production tasks such as manufacturing, in which material assets (outputs) are produced using raw materials (inputs) and manufacturing resources, facilities, and manpower (mechanisms), subject to the manufacturing rules and procedures (controls). IDEF0 function modeling has since been adopted for other applications such as business process modeling and DoDAF.

When using IDEF0 for DoDAF activity modeling, an interesting problem arises when dealing with information transfer activities. Many functions within larger processes involve activities that simply move information or products from one location or node to another. Architects may find themselves creating many of the same types of information transfer activities, some inferred and some specifically outlined in governing publications. These activities are easily found in terms such as obtain, receive, transmit, issue, distribute, submit, store, and others. We call these activities "hollow" transfer activities. They are hollow because they do not contain a transformation function that produces a new and unique output. They are transfer activities because they simply move information from one location to another. The information content is not changed or transformed in any way; it is merely transferred or made available to support other activities or functions.

The question of how an architect should show these types of activities within IDEF0 and other modeling methods generates considerable debate in the modeling community. Some IDEF0 and other modeling purists would argue that the "obtain information" activity should not be shown

FIGURE 2. **Activity Diagram Showing Hollow Activity.**

because it does not show a transformation. Others would argue that the discussion is somewhat trivial or should be left to system views, not operational activities. But there is a danger, depending on the purpose of the architecture, in leaving these activities out of operational views.

For example, Figure 2 shows two activity models depicting the same mission-tasking process. A mission objective is received, analyzed, and broken down into one or more mission tasks, which are distributed to mission planners and then used as controls to help create a mission plan. This same process could occur within a single room or in different locations across the world. The top model in the figure shows the "distribute task to planners" hollow transfer activity. The bottom model does not. In the top model, there is no doubt that the transfer activity has visibility. But again, it is hollow because there is no transformation; the output is the same as the control. The "mission task" control is also the "mission task" output. By showing the hollow transfer activity, it is very clear that the "distribute task to planners" activity must occur, and there should be a mechanism (person or technology) assigned to it. A missing mechanism could show a gap in capabilities, especially since the "distribute task to planners" activity can take significant time and resources.

The bottom model does not include the "distribute task to planners" activity. It is more precise and focuses on the core activities that are not distribution functions. As such, there are no hollow transfer activities depicted. From this model, it would be easy to overlook output distribution activities and the mechanisms that execute them. In simple terms, one could model a process but not see its distribution pitfalls and thus not ensure the right information gets to the right people. If the operational activities do not include the transfer activities, systems and their functions may not be appropriately visible. Eliminating hollow transfer activities may also not accurately capture what could be the most time- and resource-intensive activities within an enterprise.

Ultimately, if the purpose of the architecture includes ensuring the right information or product gets to the right people at the right time, or mapping existing real-world, constraint-based and location-dependent processes, it is essential that hollow transfer activities be represented in some form. That may mean altering existing modeling techniques or investigating new methods to answer the

critical questions. How an architecting team deals with this can be critical, especially considering the growing importance of interoperable and net-centric architectures. The failure to make certain activities visible within an operational architecture can influence where future investment and existing resources are spent. Failure to model transfer activities properly or make them visible for analysis can inadvertently further DoD interoperability and information-distribution problems. Therefore, it is critical to examine hollow transfer activities to prevent distribution problems and lack of interoperability.

To Wrap it Up

In summary, we have proposed the following heuristics in order to help overcome and avert problems when developing joint operational architectures:

- Lack of knowledge in the domain equals architecture pain. A readily available network of subject matter experts makes the architecture relevant.
- To have a joint vocabulary, use the joint dictionary. Seek a common understandable vocabulary by referencing joint standards and the joint dictionary.
- When defining a joint process stalls, there needs to be a process owner to make firm calls. When establishing an enterprise-wide operational architecture, there needs to be one boss to overcome irreconcilable differences across stakeholders.
- Abstract too high—the models can lie; abstract too low— one gets lost in the flow. Architect at the level of abstraction that provides the answers sought.
- For the process to rule the show, organizational bias has to go. People tend to think "organization" first, not "process," and architecture models should be created independent of the organization.
- To confine the architecture to only one level of war can make the architecture poor. Follow the process and information flows; do not limit context to operational or tactical level if not a necessary constraint.
- Critically examine hollow transfer activities to prevent distribution problems and lack of interoperability. Be critical of hollow transfer activities and ensure they have the appropriate visibility in order to prevent and address capability gaps.

As systems increase in complexity, the architect's job will continue to be tested. These simple heuristics can help increase interoperability and the gains produced from architectural development in the DoD.

The authors welcome comments and questions. Contact them at todd.wieser@hurlburt.af.mil, gregory.j.miller@pentagon.af.mil, aaron.piepkorn@pentagon.af.mil, james.kennedy@pentagon.af.mil, robert.mills@afit.edu, and john.colombi@afit.edu.